# Recursion: Fibonacci Numbers

*The Fibonacci Sequence*

The Fibonacci sequence appears in nature all around us, in the arrangement of seeds in a sunflower and the spiral of a nautilus for example.

The Fibonacci sequence begins with $fibonacci(0) = 0$ and $fibonacci(1) = 1$ as its first and second terms. After these first two elements, each subsequent element is equal to the sum of the previous two elements.

Programmatically:

- $fibonacci(0) = 0$
- $fibonacci(1) = 1$
- $fibonacci(n) = fibonacci(n-1) + fibonacci(n-2)$

Given $n$, return the $n^{th}$ number in the sequence.

As an example, $n = 5$. The Fibonacci sequence to 6 is $fs = [0, 1, 1, 2, 3, 5, 8]$. With zero-based indexing, $fs[5] = 5$.

**Function Description**

Complete the recursive function $fibonacci$ in the editor below. It must return the $n^{th}$ element in the Fibonacci sequence.

fibonacci has the following parameter(s):

- *n*: the integer index of the sequence to return

**Input Format**

The input line contains a single integer, $n$.

**Constraints**

- $0 < n \leq 30$

**Output Format**

Locked stub code in the editor prints the integer value returned by the $fibonacci$ function.

**Sample Input**

```
3
```

**Sample Output**

```
2
```

**Explanation**

The Fibonacci sequence begins as follows:

$fibonacci(0) = 0$

$fibonacci(1) = 1$
$fibonacci(2) = (0 + 1) = 1$
$fibonacci(3) = (1 + 1) = 2$
$fibonacci(4) = (1 + 2) = 3$
$fibonacci(5) = (2 + 3) = 5$
$fibonacci(6) = (3 + 5) = 8$
...

We want to know the value of $fibonacci(3)$. In the sequence above, $fibonacci(3)$ evaluates to $2$.